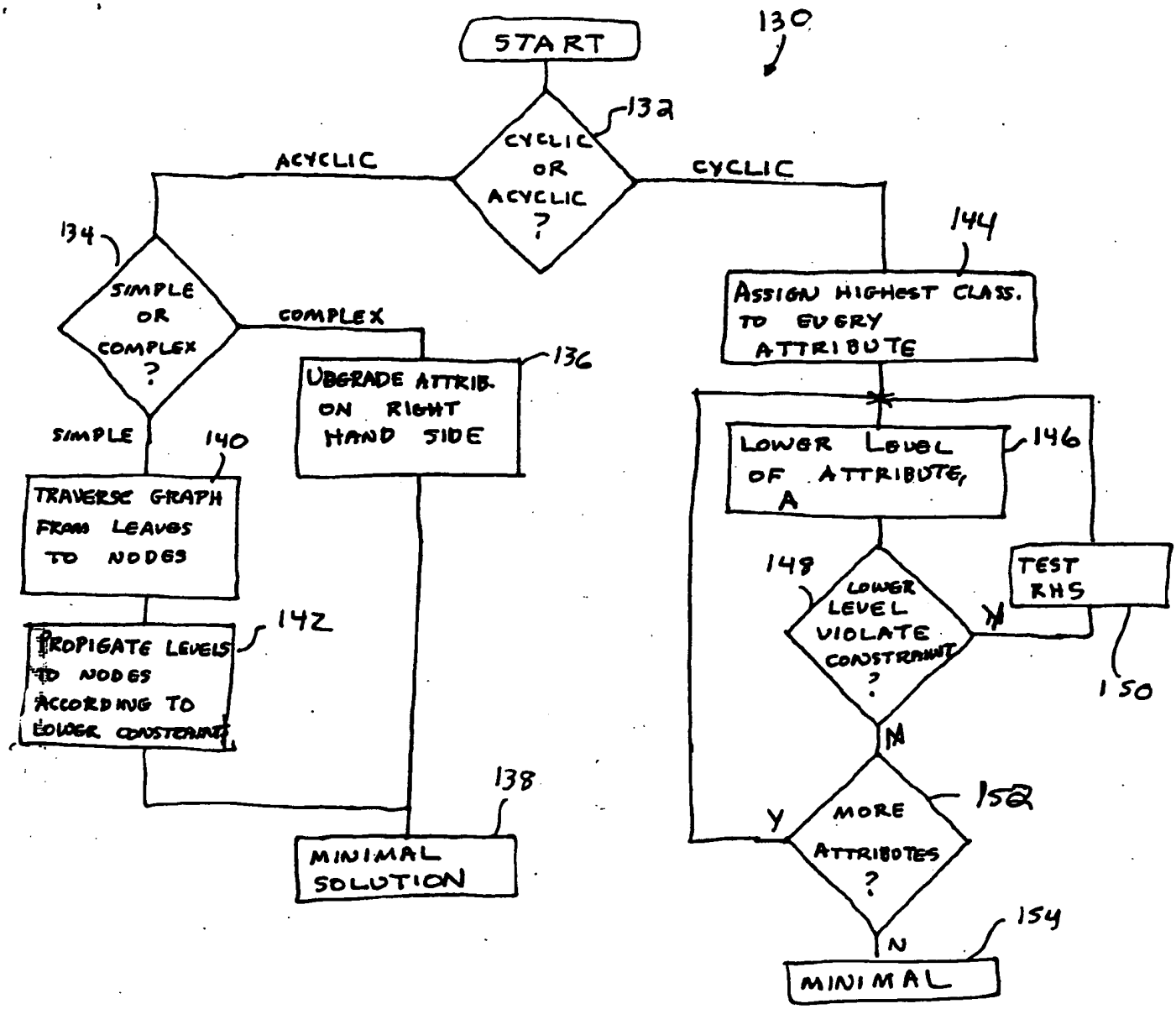
START
130
132
CYCLIC OR ACYCLIC ?
ACYCLIC
CYCLIC
134
SIMPLE OR COMPLEX ?
COMPLEX
SIMPLE
136
UPGRADE ATTRIB. ON RIGHT HAND SIDE
140
TRAVERSE GRAPH FROM LEAVES TO NODES
142
PROPIGATE LEVELS TO NODES ACCORDING TO LOWER CONSTRAINT
138
MINIMAL SOLUTION
144
ASSIGN HIGHEST CLASS. TO EVERY ATTRIBUTE
146
LOWER LEVEL OF ATTRIBUTE, A
148
LOWER LEVEL VIOLATE CONSTRAINT ?
N
TEST RHS
150
Y
152
MORE ATTRIBUTES ?
N
154
MINIMAL

FIGURE 6

$\lambda(\texttt{illness}) \succeq \text{Research}$
$\lambda(\texttt{prescription}) \succeq \text{Clinical}$
$\lambda(\texttt{prescription}) \succeq \lambda(\texttt{treatment})$
$\lambda(\texttt{treatment}) \succeq \text{Public}$
$\lambda(\texttt{treatment}) \succeq \lambda(\texttt{visit})$
$\lambda(\texttt{treatment}) \succeq \lambda(\texttt{illness})$
$\lambda(\texttt{visit}) \succeq \text{Public}$

Figure 7 (a)

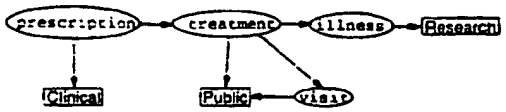

Figure 7 (b)

$\lambda(\texttt{illness}) = \text{Research}$
$\lambda(\texttt{prescription}) = \text{Clinical}$
$\lambda(\texttt{treatment}) = \text{Research}$
$\lambda(\texttt{visit}) = \text{Public}$

Figure 7 (c)

$\lambda(\texttt{patient}) \succeq \text{Public}$
$\lambda(\texttt{bill}) \succeq \text{Financial}$
$\text{lub}\{\lambda(\texttt{patient}), \lambda(\texttt{bill})\} \succeq \text{Admin}$

Figure 8 (a)

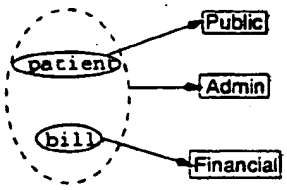

Figure 8 (b)

$\lambda(\texttt{bill}) = \text{Admin}$
$\lambda(\texttt{patient}) = \text{Public}$

Figure 8 (c)

$\lambda(\texttt{bill}) = \text{Financial}$
$\lambda(\texttt{patient}) = \text{Research}$

Figure 8 (d)

$\lambda(\texttt{division}) \succeq \text{Public}$
$\text{lub}\{\lambda(\texttt{division}), \lambda(\texttt{plan})\} \succeq \lambda(\texttt{doctor})$
$\lambda(\texttt{doctor}) \succeq \text{Public}$
$\lambda(\texttt{doctor}) \succeq \lambda(\texttt{illness})$
$\lambda(\texttt{illness}) \succeq \lambda(\texttt{division})$
$\lambda(\texttt{illness}) \succeq \text{Research}$
$\lambda(\texttt{plan}) \succeq \text{Financial}$

Figure 9 (a)

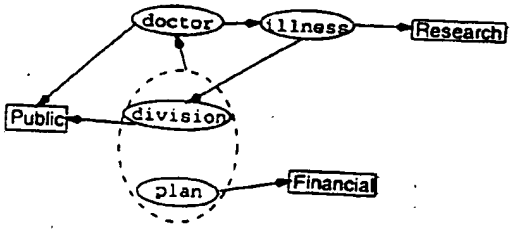

Figure 9 (b)

$\lambda(\texttt{division}) = \text{Public}$
$\lambda(\texttt{doctor}) = \text{Research}$
$\lambda(\texttt{illness}) = \text{Research}$
$\lambda(\texttt{plan}) = \text{Admin}$

Figure 9 (c)

Algorithm 3.1 (Minimal Classification Generation)

MAIN
For $A \in \mathcal{A}$ do $Constr[A] := \emptyset$; $visit[A] := 0$; $done[A] :=$ FALSE
For $l \in L$ do $done[l] :=$ TRUE; $visit[l] := 1$
For $c=(lhs,rhs) \in C_{lower}$ do
  $count[c] := 0$
  For $A \in lhs$ do
    $Constr[A] := Constr[A] \cup \{c\}$; $count[c] := count[c] + 1$
$Stack := \emptyset$
For $A \in \mathcal{A}$ do
  If $visit[A] = 0$ then dfs_visit(A)
$max\_scc := 0$
For $i = 1, \ldots, |\mathcal{A}|$ do $scc[i] := \langle\rangle$
For $A \in \mathcal{A}$ do $visit[A] := 0$
While NOTEMPTY($Stack$) do
    $A :=$ POP($Stack$)
    If $visit[A] = 0$ then
      $max\_scc := max\_scc + 1$
      $scc[max\_scc] := \langle A \rangle$
      dfs_back_visit(A)
For $A \in \mathcal{A}$ do $\lambda(A) := \top$; $visit[A] := 0$
compute_upper_bounds
compute_partial_lubs
compute_minimal_solution

COMPUTE_UPPER_BOUNDS
For $(l, A) \in C_{upper}$ do $\lambda(A) := \lambda(A) \sqcap l$
For $i := 1, \ldots, max\_scc$ do
  For $A \in scc[i]$ do
    If $visit[A] = 0$ then upper_bound(A, i)

UPPER_BOUND(A, i)
$visit[A] := 1$
For $c = (lhs, rhs) \in Constr[A]$ do
  If $count[c] > 0$ then $count[c] := count[c] - 1$
  If $count[c] = 0$ or $rhs \in scc[i]$ then
    $levlhs := \bot$
    For $A' \in lhs$ do $levlhs := levlhs \sqcup \lambda(A')$
    If $\neg(levlhs \succeq \lambda(rhs))$ then
      If $rhs \in L$ then Fail
      else $\lambda(rhs) := \lambda(rhs) \sqcap levlhs$
          If $rhs \in scc[i]$ then
            upper_bound(rhs, i)

COMPUTE_MINIMAL_SOLUTION
For $i := max\_scc, \ldots, 1$ do
  For $A \in scc[i]$ do
    $done[A] :=$ TRUE; $l := \bot$
    For $c=(lhs,rhs) \in Constr[A]$ do
      If $done[rhs]$ then
        case $|lhs|$ of
          1: $l := l \sqcup \lambda(rhs)$
          >1: $l := l \sqcup$ minlevel(A, c)
      else $done[A] :=$ FALSE
    If $done[A]$ then $\lambda(A) := l$
    else $DSet := \{l' \mid l'$ is a maximal level, $\lambda(A) \succ l' \succeq l\}$
        While $DSet \neq \emptyset$
          Choose l'' in $DSet$; $DSet := DSet - l''$
          $Lower :=$ try_to_lower(A, l'')
          If $Lower \neq \emptyset$ then
            For $(A', l') \in Lower$ do $\lambda(A') := l'$
            $DSet := \{l' \mid l'$ maximal level, $\lambda(A) \succ l' \succeq l\}$
        $done[A] :=$ TRUE
    For $c \in Constr[A]$ do
      $j := count[c]$; $Plub[c][j] := \lambda(A) \sqcup Plub[c][j+1]$
      $count[c] := count[c] - 1$

DFS_VISIT(A)
$visit[A] := 1$
For $(lhs, rhs) \in Constr[A]$ do
  If $visit[rhs] = 0$ then dfs_visit(rhs)
PUSH(A,$Stack$)

DFS_BACK_VISIT(A)
/* Traverses the constraints backward and inserts all attributes found in the same SCC list as A */
$visit[A] := 1$
For $(lhs, A) \in C_{lower}$ do
  For $A' \in lhs$ do
    If $visit[A'] = 0$ then
      $scc[max\_scc] :=$ concat($\langle A' \rangle$, $scc[max\_scc]$)
      dfs_back_visit(A')

COMPUTE_PARTIAL_LUBS
For $c=(lhs,rhs) \in C_{lower}$ do $count[c] := 0$; $Plub[c][0] := \bot$
For $i := 1, \ldots, max\_scc$ do
  For $A \in$ reverse($scc[i]$) do
    For $c = (lhs, rhs) \in Constr[A]$ do
      $count[c] := count[c] + 1$; $j := count[c]$
      $Plub[c][j] := Plub[c][j-1] \sqcup \lambda(A)$
For $c=(lhs,rhs) \in C_{lower}$ do $j := count[c] + 1$; $Plub[c][j] := \bot$

MINLEVEL(A,c)
/* Returns a minimal level for A that keeps c satisfied */
$j := count[c]$; $(lhs, rhs) := c$; $last := \lambda(A)$
$lubothers := Plub[c][j-1] \sqcup Plub[c][j+1]$
If $lubothers \succeq \lambda(rhs)$ then $last := \bot$
else $Try := \{l \mid l$ is a maximal level s. t. $last \succ l\}$
    While $Try \neq \emptyset$ do
      Choose l in Try; $Try := Try - l$
      If $(l \sqcup lubothers) \succeq \lambda(rhs)$ then
        $last := l$; $Try := \{l \mid l$ is a maximal level s. t. $last \succ l\}$
return $last$

TRY_TO_LOWER(A,l)
$Tocheck := \{(A, l)\}$
$Tolower := \emptyset$
Repeat
  Choose $(A', l') \in Tocheck$
  $Tocheck := Tocheck - \{(A', l')\}$
  $Tolower := Tolower \cup \{(A', l')\}$
  For $(lhs, rhs) \in Constr[A']$ do
    $level := \bot$
    For $A'' \in lhs$ do
      If $\exists (A'', l'') \in Tolower$ then
        $level := level \sqcup l''$
      else $level := level \sqcup \lambda(A'')$
    case $done[rhs]$ of
      TRUE: If $\neg(level \succeq \lambda(rhs))$ then return $\emptyset$
      FALSE: If $\neg(level \succeq \lambda(rhs))$ then
          $newlevel := \lambda(rhs) \sqcap level$
          If $\exists (rhs, l'') \in (Tolower \cup Tocheck)$ then
            If $\neg(newlevel \succeq l'')$ then
              $newlevel := l'' \sqcap newlevel$
              If $(rhs, l'') \in Tolower$ then
                $Tolower := Tolower - \{(rhs, l'')\}$
              else $Tocheck := Tocheck - \{(rhs, l'')\}$
              $Tocheck := Tocheck \cup \{(rhs.\ newlevel)\}$
          else $Tocheck := Tocheck \cup \{(rhs.\ newlevel)\}$
until $Tocheck = \emptyset$
return $Tolower$

Figure 10 Algorithm for computing a minimal classification.

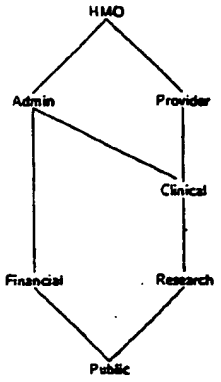

Figure 11 (a)

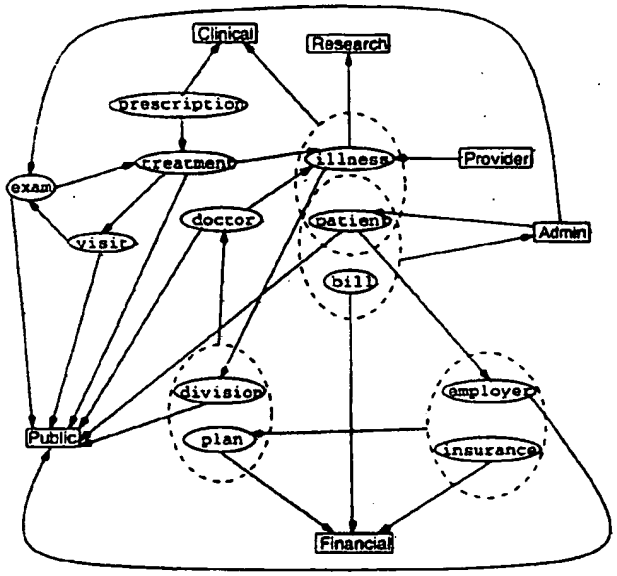

Figure 11 (b)

SCC

	• initial levels	[8] doctor	[8] division	[8] illness	[7] plan	[6] employer	[5] patient	[4] bill	[3] insurance	[2] exam	[2] treatment	[2] visit	[1] prescription
		HMO	HMO	HMO	HMO	HMO	HMO	HMO	HMO	HMO	HMO	HMO	HMO
doctor	compute_upper_bounds	HMO	Clinical	Clinical	HMO	Admin	Admin	HMO	HMO	Admin	Admin	Admin	HMO
	try_to_lower(doctor,Admin)	Admin	Clinical	Clinical	HMO	Admin	Admin	HMO	HMO	Admin	Admin	Admin	HMO
	try_to_lower(doctor,Financial) F	Admin	Clinical	Clinical	HMO	Admin	Admin	HMO	HMO	Admin	Admin	Admin	HMO
	try_to_lower(doctor,Clinical)	Clinical	Clinical	Clinical	HMO	Admin	Admin	HMO	HMO	Admin	Admin	Admin	HMO
	try_to_lower(doctor,Research)	Research	Research	Research	HMO	Admin	Admin	HMO	HMO	Admin	Admin	Admin	HMO
	try_to_lower(doctor,Public) F	Research	Research	Research	HMO	Admin	Admin	HMO	HMO	Admin	Admin	Admin	HMO
division	–		Public	Research	HMO	Admin	Admin	HMO	HMO	Admin	Admin	Admin	HMO
illness	–			Research	HMO	Admin	Admin	HMO	HMO	Admin	Admin	Admin	HMO
plan	–				Admin	Admin	Admin	HMO	HMO	Admin	Admin	Admin	HMO
employer	–					Public	Admin	HMO	HMO	Admin	Admin	Admin	HMO
patient	–						Clinical	HMO	HMO	Admin	Admin	Admin	HMO
bill	–							Financial	HMO	Admin	Admin	Admin	HMO
insurance	–								Admin	Admin	Admin	Admin	HMO
exam	try_to_lower(exam,Financial) F									Admin	Admin	Admin	HMO
	try_to_lower(exam,Clinical)									Clinical	Clinical	Clinical	HMO
	try_to_lower(exam,Research)									Research	Research	Research	HMO
	try_to_lower(exam,Public) F									Research	Research	Research	HMO
treatment	–										Research	Research	HMO
visit	–											Research	HMO
prescription	–												Clinical
	• final levels	Research	Public	Research	Admin	Public	Clinical	Financial	Admin	Research	Research	Research	Clinical

Figure 11 (c)

		SCC											
		[8]			[7]	[6]	[5]	[4]	[3]	[2]			[1]
	• initial levels	doctor	division	illness	plan	employer	patient	bill	insurance	exam	treatment	visit	prescription
		HMO	HMO	HMO	HMO	HMO	HMO	HMO	HMO	HMO	HMO	HMO	HMO
patient	compute_upper_bounds	HMO	Clinical	Clinical	HMO	Admin	Admin	HMO	HMO	Admin	Admin	Admin	HMO
	try_to_lower(patient,Financial)	HMO	Clinical	Clinical	HMO	Financial	Financial	HMO	HMO	Admin	Admin	Admin	HMO
	try_to_lower(patient,Public)	HMO	Clinical	Clinical	HMO	Public	Public	HMO	HMO	Admin	Admin	Admin	HMO
plan	try_to_lower(plan,Admin)	Admin	Clinical	Clinical	Admin	Public	Public	HMO	HMO	Admin	Admin	Admin	HMO
	try_to_lower(plan,Financial)	Admin	Clinical	Clinical	Financial	Public	Public	HMO	HMO	Admin	Admin	Admin	HMO
doctor	try_to_lower(doctor,Financial)F	Admin	Clinical	Clinical	Financial	Public	Public	HMO	HMO	Admin	Admin	Admin	HMO
	try_to_lower(doctor,Clinical)	Clinical	Clinical	Clinical	Financial	Public	Public	HMO	HMO	Admin	Admin	Admin	HMO
	try_to_lower(doctor,Research)F	Clinical	Clinical	Clinical	Financial	Public	Public	HMO	HMO	Admin	Admin	Admin	HMO
division	-	Clinical	Research	Clinical	Financial	Public	Public	HMO	HMO	Admin	Admin	Admin	HMO
illness	-				Financial	Public	Public	HMO	HMO	Admin	Admin	Admin	HMO
employer	-					Public		HMO	HMO	Admin	Admin	Admin	HMO
bill	-							Admin	HMO	Admin	Admin	Admin	HMO
insurance	-								Financial	Admin	Admin	Admin	HMO
exam	try_to_lower(exam,Financial)F									Admin	Admin	Admin	HMO
	try_to_lower(exam,Clinical)									Clinical	Clinical	Clinical	HMO
	try_to_lower(exam,Research)F									Clinical	Clinical	Clinical	HMO
treatment	try_to_lower(treatment,Clinical)										Clinical	Clinical	HMO
visit	-											Clinical	HMO
prescription	-												Clinical
	• final levels	Clinical	Research	Clinical	Financial	Public	Public	Admin	Financial	Clinical	Clinical	Clinical	Clinical

Figure 12

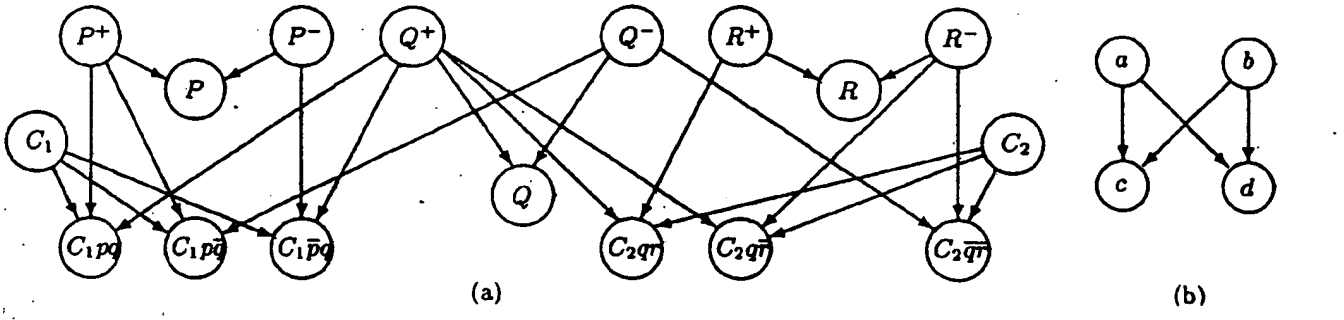

Figure 13